What is claimed is:

1. A method for updating a protected partition within a hard drive of a computing system, wherein said method comprises:

starting execution of an initialization program in a processor within said computing system in response to turning on electrical power within said computing system;

determining whether an update partition file is stored in non-volatile storage within said computing system for subsequently updating said protected partition;

after determining that said update partition is stored within said computing system for updating said protected partition, writing a portion of said update partition file to said protected partition; and

locking said protected partition to prevent further modification of information stored within said protected partition.

2. The method of claim 1, wherein

a flag bit is set in non-volatile storage within said computing system when said update partition file is stored in non-volatile storage within said computing system, and

determining whether said update partition is stored within said computing system for updating said protected partition is performed by determining whether said flag bit is set.

3. The method of claim 1, wherein

said method additionally comprises, after determining that said update partition file is stored within said computing system for updating said protected

partition, verifying whether said update partition file has been generated by a trusted server system, and

said portion of said update partition is written to said protected partition only following verification that said update partition file has been generated by a trusted server system.

4. The method of claim 3, wherein verification that said update partition file has been generated by said trusted server system includes:

forming a first message digest by applying a hash algorithm to a portion of said update partition file;

forming a second message digest by decrypting a digital signature within said update partition file using a public key of said trusted server system; and;

determining that said first and second message digests are identical.

5. The method of claim 3, wherein

a setup password is stored in non-volatile storage within said computing system,

verifying that said update partition file has been generated by said trusted server system includes signing an encrypted portion of said update partition file with a public key of said trusted server system, and

said encrypted portion of said update partition file has been prepared by signing, with a private key of said trusted server system, a result of the application of an algorithm to data including a version of said setup password accessed by said trusted server system.

6. The method of claim 5, wherein

said data includes said version of said setup password appended to a portion of said update partition file,

said algorithm is a hash algorithm generating a message digest, and

verifying that said update partition file has been generated by said trusted server system includes applying said hash algorithm to said setup password stored within said computing system appended to a portion of said update partition file to generate a first version of a message digest and comparing said first version of said message digest with a second version of said message digest obtained by signing said encrypted portion of said update partition file.

7. The method of claim 1, wherein

said update partition file includes a plurality of entries and a plurality of encrypted elements,

each encrypted element within said plurality of encrypted elements is associated with an entry in said plurality of entries.

said method additionally comprises, following determining that said update partition file is stored within said computing system for updating said protected partition, verifying whether each entry in said plurality of entries within said update partition file has been generated by a trusted server system, and

each entry in said plurality of entries within said update partition is written to said protected partition only following verification that said entry has been generated by a trusted server system.

8. The method of claim 7, wherein verifying that said entry has been generated by said trusted server system includes:

forming a first message digest by applying a hash algorithm to said entry;

forming a second message digest by signing said encrypted element associated with said entry using a public key of said trusted server system; and;

determining that said first and second message digests are identical.

9. The method of claim 7, wherein

a setup password is stored in non-volatile storage within said computing system,

verifying that said entry has been generated by said trusted server system includes signing said encrypted element associated with said entry with a public key of said trusted server system, and

said encrypted element of said update partition file has been prepared by signing, with said private key of said trusted server system, a result of the application of an algorithm to data including a version of said setup password accessed by said trusted server system.

10. The method of claim 9, wherein

said data includes said version of said setup password appended to a said entry,

said algorithm is a hash algorithm generating a message digest, and

verifying that said entry has been generated by said trusted server system includes applying said hash algorithm to said setup password stored within said computing system appended said entry to generate a first version of a message digest and comparing said first version of said message digest with a second version of said message digest obtained by signing said encrypted element.

11. The method of claim 7, wherein

information stored in said protected partition is compared to each entry in said plurality of entries within said update partition,

when a matching portion of said information stored in said protected partition is found to be similar to said entry, said matching portion is overwritten with said entry if space around said matching portion is sufficient, and

when a matching portion of said information stored in said protected partition is not found to be similar to said entry, said entry is appended to said information stored in said protected partition if space within said protected partition is sufficient.

12. The method of claim 1, wherein

said method additionally comprises receiving an input signal from a keyboard of said computing system and comparing said input signal with a signal corresponding to a setup password stored in non-volatile storage within said computing system, and

said protected partition is left unlocked if said input signal matches said signal corresponding to said setup password.

13. A method for updating a protected partition within a hard drive of a client computing system, wherein said method comprises:

generating an update partition file within a server;

transferring said update partition file from said server to said client computing system;

storing said update partition file in non-volatile storage within said client computing system;

starting execution of an initialization program in a processor within said client computing system in response to turning on electrical power within said client computing system;

determining that said update partition file is stored in non-volatile storage within said client computing system;

writing a portion of said update partition file to said protected partition; and

locking said protected partition to prevent further modification of information stored within said protected partition.

14. The method of claim 13, wherein said update partition file is transferred from said server to said client computing system by means of electrical signals transmitted through a public switched telephone network.

15. The method of claim 13, wherein said update partition file is transferred from said server to said client computing system by means of electrical signals transmitted over a local area network.

16. The method of claim 13, wherein transferring said update partition file from said server to said client computing system includes:

writing said update partition file to a removable computer readable medium from said server;

transporting said removable computer readable medium from said sever to said client computing system; and

reading said update partition file from said removable computer readable medium into said client computing system.

17. The method of claim 13, wherein

a flag bit is set in non-volatile storage within said client computing system when said update partition file is stored in non-volatile storage within said client computing system, and

determining that said update partition file is stored in non-volatile storage within said client computing system includes determining that said flag bit is set.

18. The method of claim 13, wherein

said method additionally comprises, following a determination that said update partition file is stored within said client computing system for updating said protected partition, verifying within said client computer system that said update partition file has been generated by said server, and

said portion of said update partition is written to said protected partition only following verification that said update partition file has been generated by said server.

19. The method of claim 18, wherein:

generating said update partition file within said server includes forming a first message digest by applying a hash algorithm to a portion of said update partition file, signing said first message digest with a private key of said server to form a digital signature, and appending said digital signature to data within said update partition file; and

verifying within said client computing system that said update partition file has been generated by said server includes forming a second message digest by applying a hash algorithm to a portion of said update partition file, forming a third message digest by signing said digital signature within said update partition

file using a public key of said server, and determining that said second and third message digests are identical.

20. The method of claim 18, wherein:

a setup password is stored in non-volatile storage within said client computing system;

a copy of said setup password is stored in a database accessible to said server;

generating said update partition file within said server includes forming an encrypted portion of said update partition file by signing a result of the application of an algorithm to data including said copy of said setup password; and

verifying within said client computing system that said update partition file has been generated by said server includes signing said encrypted portion of said update partition file with a public key of said server.

21. The method of claim 20, wherein

said data includes said version of said setup password appended to a portion of said update partition file,

said algorithm is a hash algorithm generating a message digest, and

verifying within said client computing system that said update partition file has been generated by said trusted server includes applying said hash algorithm to said setup password stored within said client computing system appended to a portion of said update partition file to generate a first version of a message digest and comparing said first version of said message digest with a second version of said message digest obtained by signing said encrypted portion of said update partition file with said public key of said server.

22. The method of claim 13, wherein

said update partition file includes a plurality of entries and a plurality of encrypted elements,

each encrypted element within said plurality of encrypted elements is associated with an entry in said plurality of entries.

said method additionally comprises, following a determination that said update partition file is stored within said client computing system for updating said protected partition, verifying within said client computing system whether each entry in said plurality of entries within said update partition file has been generated by a server, and

each entry in said plurality of entries within said update partition is written to said protected partition only following verification that said entry has been generated by said server.

23. The method of claim 22, wherein

each said encrypted element is formed in said server by applying a hash algorithm to said entry, forming a first message digest, and by signing said first message digest with a private key of said server; and

verification that said entry has been generated by said server includes forming a second message digest by applying a hash algorithm to said entry, forming a third message digest by signing said encrypted element associated with said entry using a public key of said server, and determining that said second and third message digests are identical.

24. The method of claim 22, wherein

a setup password is stored in non-volatile storage within said client computing system;

a copy of said setup password is stored in a database accessed by said server;

said encrypted element of said update partition file is prepared in said server by signing, with a private key of said server, a result of the application of an algorithm to data including said copy of said setup password; and

verification within said client computing system that said entry has been generated by said server includes signing said encrypted element associated with said entry with said public key of said server,

25. The method of claim 24, wherein

said data includes said version of said setup password appended to a said entry,

said algorithm is a hash algorithm generating a message digest, and

said verification that said entry has been generated by said server includes applying said hash algorithm to said setup password stored within said client computing system appended to said entry to generate a first version of a message digest and comparing said first version of said message digest with a second version of said message digest obtained by signing said encrypted element.

26. A computer system comprising:

a processor executing an initialization program in response to power being turned on in said computer program;

a hard drive having a protected partition blocked during execution of an initialization program to prevent changing information stored within said protected partition;

non-volatile storage storing an update partition data structure for modifying contents of said protected partition and said initialization program, wherein said initialization program executing within said processor determines that said update partition data structure is stored in said non-volatile storage, writes a portion of said update partition data structure to said protected partition, and locks said protected partition to prevent further modification of information stored within said protected partition.

27. The computer system of claim 26, wherein

a flag bit is set in non-volatile storage within said computing system when said update partition data structure is stored in non-volatile storage within said computing system, and

said initialization program determines said update partition is stored within said computing system for updating said protected partition is performed by determining that said flag bit is set.

28. The computer system of claim 26, wherein

after determining that said update partition data structure is stored within said computing system for updating said protected partition, said initialization program verifies whether said update partition data structure has been generated by a trusted server system, and

said portion of said update partition is written to said protected partition only following verification that said update partition data structure has been generated by a trusted server system.

29. The computer system of claim 28, wherein

said update partition data structure includes a plurality of entries and a plurality of encrypted elements,

each encrypted element within said plurality of encrypted elements is associated with an entry in said plurality of entries, and

said initialization program uses each said encrypted element to determine that an entry associated with said encrypted element has been generated by said trusted server system.

30. The computer system of claim 29, wherein

said non-volatile storage additionally stores a setup password, and

each said encrypted element includes a digital signature signed by said trusted server system, wherein said digital signature is formed by applying a hash algorithm to an entry associated with said encrypted element to form a message digest and by signing said message digest with a private key of said trusted server system.

31. A computer-readable medium, having stored thereon a data structure comprising a plurality of entries and a plurality of encrypted elements, wherein

each encrypted element within said plurality of encrypted elements is associated with an entry in said plurality of entries, and

each said encrypted element includes a digital signature signed by a trusted server system, wherein said digital signature is formed by applying a hash algorithm to an entry associated with said encrypted element, appended with a setup password of said computer system to form a message digest and by signing said message digest with a private key of said trusted server system.

32. A system for updating a protected partition within a hard drive of a remote computing system, wherein said system comprises:

a server including a database storing a setup password of said remote computer system and a public key of said remote computer system, and storage having stored thereon a data structure comprising a plurality of entries and a plurality of encrypted elements, wherein each encrypted element within said plurality of encrypted elements is associated with an entry in said plurality of entries, and each said encrypted element includes a digital signature signed by said server, wherein said digital signature is formed by applying a hash algorithm to an entry associated with said encrypted element to form a message digest and by signing said message digest with a private key of said server;

means for transferring said data structure from said server to said remote computing system;

a processor within said remote computer system;

non-volatile storage within said remote computer system storing an initialization program for execution within said processor in response to power being turned on within said remote computer system, wherein said initialization

program executing within said processor determines that said update partition data structure is stored in said non-volatile storage, determines that each entry has been generated by said server, writes a portion of said entries to said protected partition, and locks said protected partition to prevent further modification of information stored within said protected partition.